RESEARCH

17(47), 2020

ISSN 2319–7757 EISSN 2319–7765

Indian Journal of Engineering

The study of Robot hand path planning using GA with multi-Space Obstacles

Muhammad Usman shoukat[1,2], **Yang Yiqiang**[1,2✉], **Saqib Ali Nawaz**[3,4], **Uzair Aslam Bhatti**[5,6], **Raza Muhammad Ahmad**[3,4], **Muhammad Usama Raza**[7]

[1]School of Automation and Information Sichuan University of Science and Engineering, Yibin 644000, china
[2]Artificial Intelligence Key Laboratory of Sichuan province, Sichuan University of Science and Engineering, yibin 644000, china
[3]College of Information Science and Technology, Hainan University, Haikou, China 570228
[4]State Key Laboratory of Marine Resource Utilization in the South China Sea, Hainan University, Haikou 570228, China
[5]Key Laboratory of Virtual Geographic Environment, Ministry of Education and Jiangsu Provincial Key Laboratory of NSLSCS, Nanjing Normal University, No. 1 Wenyuan Road, Nanjing, China
[6]School of Geographical Information System, Nanjing Normal University
[7]School of Materials Science and Engineering, Changzhou University, Changzhou 213000, china

Muhammad Usman shoukat – Email: usmanryk12@gmail.com
Yang Yiqiang – Email: yqyang_1992@126.com
Saqib Ali Nawaz – Email: saqibsial20@gmail.com
Uzair Aslam Bhatti – Email: uzairaslambhatti@hotmail.com
Raza Muhammad Ahmad – Email: wenbo1147@yahoo.com
Muhammad Usama Raza – Email: usama132741.raza@gmail.com

✉**Corresponding author**
School of Automation and Information Sichuan University of Science and Engineering, Yibin 644000, china / Artificial Intelligence Key Laboratory of Sichuan province, Sichuan University of Science and Engineering, yibin 644000, china
Email: yqyang_1992@126.com

Article History
Received: 30 September 2019
Accepted: 27 November 2019
Published: January 2020

Citation
Muhammad Usman shoukat, Yang Yiqiang, Saqib Ali Nawaz, Uzair Aslam Bhatti, Raza Muhammad Ahmad, Muhammad Usama Raza. The study of Robot hand path planning using GA with multi-Space Obstacles. *Indian Journal of Engineering*, 2020, 17(47), 81-96

General Note

Article is recommended to print as color digital version in recycled paper.

ABSTRACT

Robot manipulator route planning can be defined as the process of simplifying any motor task into a single movement, while satisfying the movement restriction and optimizing the cost function. Although various aspects of robotic arms path planning have been studied, the problem of rapid collision avoidance planning for work in a dynamic environment has not been resolved. This article is dedicated to developing a comprehensive, computable, managed, non-collision pathway for robotic weapons to describe path segments that connect the start, intermediate, and end points. This study presents a new, effective GA multitasking technique for two-dimensional (2D) scheduling of obstacle avoidance trajectories. A method of optimization of point-to-point path planning with the help of a genetic algorithm is proposed to solve the problem of avoiding the obstacles of three-way redundant work in two-dimensional space. The objective function in GA is set to minimize traveling time and path with their weights allocated according to their worth in actual situation. The constraint of GA is not to exceed the maximum torque limit while avoiding collisions with several obstacles in robot workspace. Forward kinematics, inverse kinematics, and polynomial path planning strategy of 3R robot manipulator designed. Route planning is designed for all types of robots, different types of connections, and dealing with different cost functions such as time and distance. In general, if there is a viable path between any two different configurations, the algorithm can generate and find it. Matlab software simulates different scenarios for a robot with different types of obstacles in a given start and target configuration. And the simulation results indicate that the developed collision-free path planning is effective which improve the traveling time and traveling distance of the manipulator.

Keywords: Manipulator, Obstacle Avoidance, Path Planning, Genetic Algorithm.

1. INTRODUCTION

Robots have the potential to improve efficiency, safety, and convenience of human endeavors. To realize this potential, it will be expedient to investigate the broad range of important disciplines in the area of robot mechanics. One of such important disciplines is the Robot's path planning and control. Robot path planning is the task of computing collision-free motions for a robotic system from a start to a goal configuration; path planning has a rich and varied history. More recent studies such as [1] proposed ant colony algorithm to find the optimal path from an initial to a final position in the presence of five obstacles [2] Using the free segment and turning point algorithms to solve obstacle introduction and path planning in known environments, the algorithm can solve two different goals, namely path security and path length. However, in practice, one problem is that there is often no complete understanding of the environment. In most cases, a detailed map with all the obstacles may not be realistic. Also, the total change in environment for the robot seems to be a difficult situation to deal with. As a result, path planning remains a fundamental problem in robotics whereby one seeks to compute a dynamically feasible trajectory to achieve a goal.

The visibility map can be described as a position map, usually for a set of points and obstacles in the Euclid plane [3]. Each node in the diagram represents the location of a point, and each edge represents a visible link between them. That is, if a line segment connecting two positions (start and end points) does not pass through any obstacles, an edge is drawn between them. When a group of positions is in a row, it can be understood as an ordered row. Therefore, the visibility map has been extended to the field of time series analysis (figure 1).

Visibility maps provide the shortest route, but they also have some drawbacks.

- Tries to stay as close as possible to obstacles
- Any execution error will lead to a collision
- Complicated in >> 2 dimensions

The unit decomposition method provides an idea to identify the geometric regions or elements of a free object. The basic algorithm for decomposing cellular path planning can be summarized as follows: First, assign "work" to the relevant area called "unit". Then find out which ones are adjacent and develop an "availability schedule." Third, identify the cells in which the basic settings and target settings are located, and scan the paths in the accessibility chart to merge the primary and target cells. Finally, in cell groupings found using appropriate search algorithms, cell breakdown is the most important aspect of cell decomposition

methods [4]. If the boundary is a function of the dielectric structure, and thus the decomposition is lossless, the method is called precise cell decay, and it also describes the exact critical curve based on the exact cell decomposition algorithm of the linear segment operation. Move freely between polygon obstacles [5].

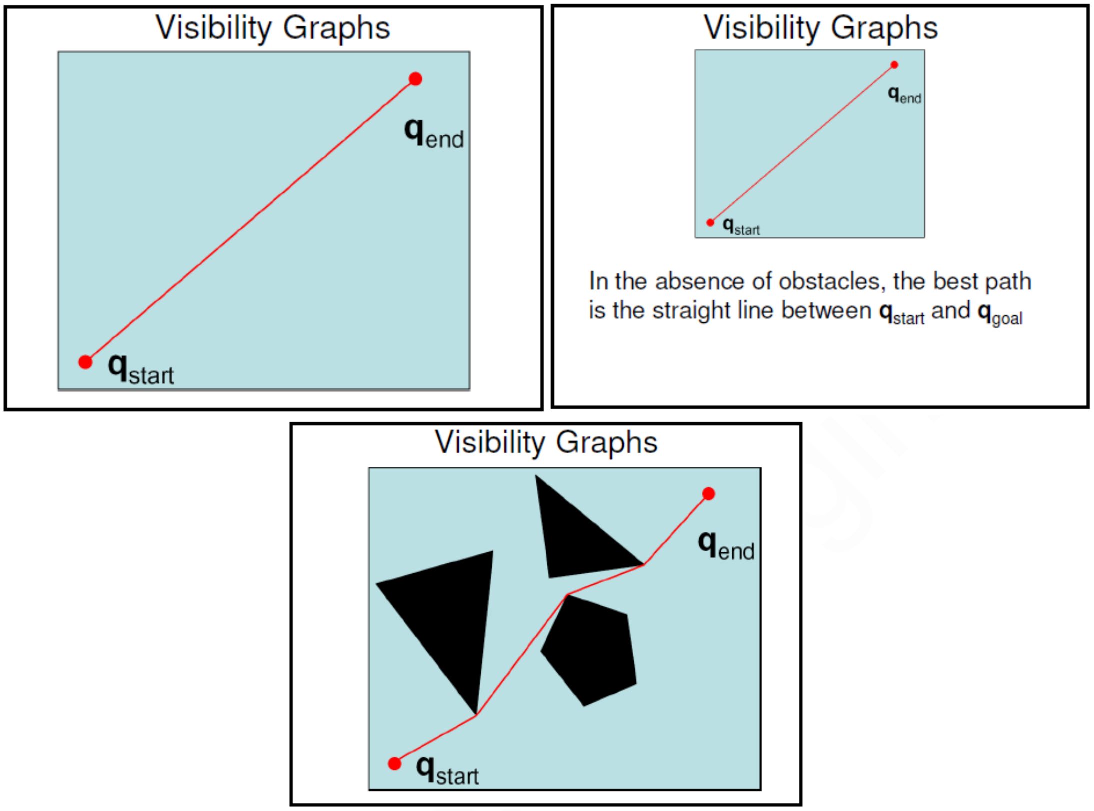


Figure 1 Visibility graph

Introduce the most popular methods to plan approximate battery scheduling [6]. The method includes constructing a sequential decomposition of the workspace as a rectangular unit and finding a communication map constructed at each decomposition level of the path. The cells are broken down. The following is possible unit decomposition similar to approximate unit decomposition. Probabilistic cell decomposition is introduced by [7] and then modified version of probabilistic presented with harmonic functions [8].

Genetic algorithms

The genetic algorithm is a search algorithm based on the natural selection and natural genetic mechanisms described [9]. They combine the survivability of the best string structure with structured but random information sharing to form a search algorithm with some innovative human search consciousness. They will try to achieve new standards of good activity from time to time. Although random genetic algorithms are not just walking around.

Binary representations are often used to encode parameters that need to be optimized. Genetic algorithm is a method based on natural selection to solve finite and infinite optimization problems. Google Analytics continually changes the settings of individual decisions. At each step, Genes selects random people from the current parent population and uses them to recruit children for the next generation. In the next generation, the population “evolved” to the best solution.

Genetic algorithm makes every step [10].

- *Selection rules* involve selections called ‘*parents*’ that contribute to the population at the next generation.
- *Crossover rules* bring together two parents to form ‘children’ for the next generation.
- *Mutation rules* apply random changes to individual parents to form children

2. METHODOLOGY

Implementation of the proposed method

In this paper the methodology for designing a collision-avoiding path planning for robot arms in obstacle environments based on genetic algorithm is presented. It turns out that everyone in the evolutionary computing community does not have a clear definition of "genetic algorithms," which distinguishes genetic algorithms from other evolutionary computational methods. However, it can be said that most methods called "GA" have at least the following elements: chromosome population, suitability selection, hybridization of new offspring, and random mutations of new offspring. Route planning uses direct kinematics to avoid singularity problems. The trajectory parameters are directly encoded using real coding as a string (chromosome) used by the genetic algorithm [11]. For three linked robots, nine parameters should be optimized. GA most often needs a fitness function that assigns a score (fitness) to each chromosome in the current population. The adaptability of chromosomes depends on the extent to which the chromosome solves the problem [12].

Each iteration of this process is called a generation. Generally, GA can be repeated for 50 to 500 generations or more [13]. The whole generation is called running. At the end of the run, there are usually one or more highly suitable chromosomes in the population. Since randomness plays an important role in each cycle, two runs using different random number seeds tend to give different detailed behavior. GA researchers typically report statistics on average across many different GA runs for each problem (for example, the best fit found on the run and the generation of the person with the best fit) [14]. The simple procedure just described is the basis for most gas applications. Some details, such as the number of populations and the possibility of hybridization and mutation, need to be filled in. The success of the algorithm usually depends on these details.

Robot hand kinematics, dynamics, and path planning strategy

Problem statement

The robot manipulator with 3 DOF must execute the task from an initial to a final position without collision with any obstacle as soon as possible. The Figure 2 shows the robot environment with 4 obstacles with circular shapes. Since each joint has its maximum torque limit, the problem is optimization to find the shortest path and traveling time under the constraints of maximum torque limit and obstacles avoidance.

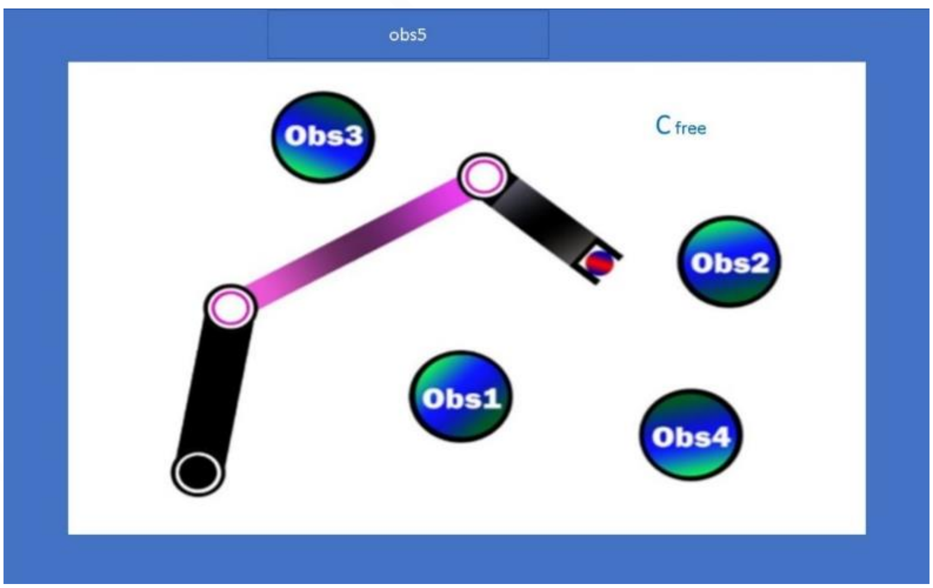


Figure 2 Robot environment with 4 obstacles with circular shapes

The proposed method has been implemented on the MATLAB platform as it has been widely accepted by scientific purposes and industrial companies. The general process for implementing an input algorithm is as follows:

- Robot and links initialization
- Environment and obstacles initialization
- Collision detection algorithm
- Kinematics modeling

- Genetic operators

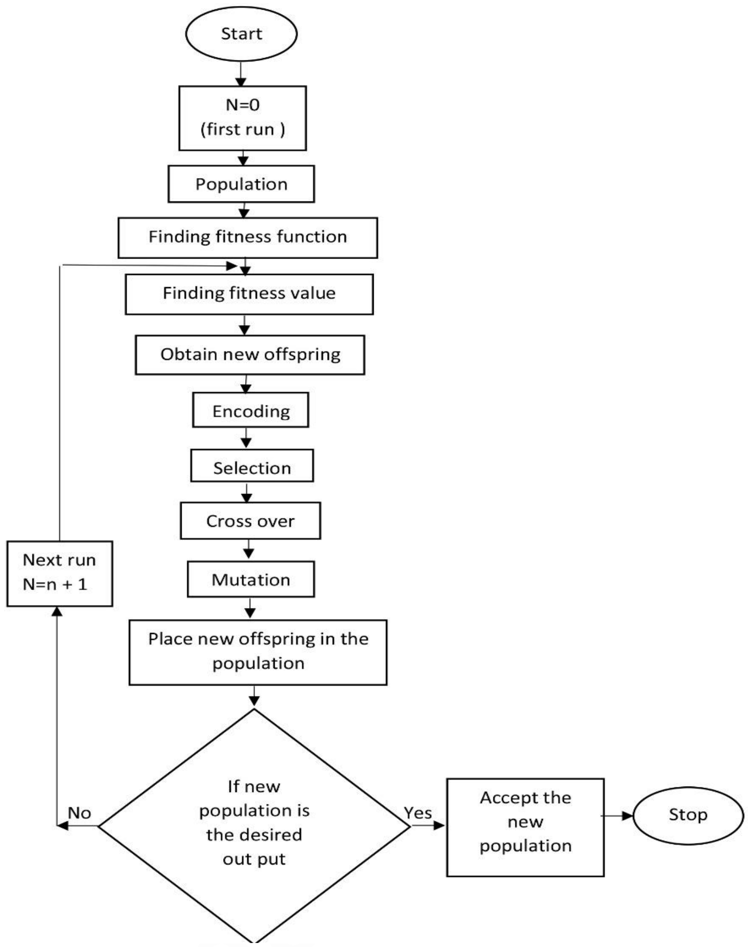


Figure 3 Genetic Algorithm flow chart

The initial population of chromosomes can be generated at random in the predefined interval of each chromosome, and the search is then carried out among this population [15]. The predominant operator adopted in the GA, are reproduction, crossover, and mutation. The copy operator is used to check the eligibility of a string (path) to be in the mating pool. A solution with excessive fitness will get a higher wide variety of copies in the mating pool; whereas a solution of low fitness can also or can also no longer have copies in the mating pool. The fitness stage required to enter the mating pool is calculated primarily based on the two ration of string fitness to average fitness of the population. The crossover operator is utilized to two cross-sites which are randomly selected. The crossover operator adopted the single factor approach with a given chance Pc, therefore, the crossover factor will be allowed for solely one gene randomly selected, in different words, the crossover operator will not disrupt the other genes. The mutation operator refers to alternation of personality values in individual string with a given probability P_m. The objective of mutation is to keep, variety in population. Now these consequences are fed to the Genetic Algorithm for producing the populace of chromosomes having optimized values (figure 3).

Start Generate random population of n chromosomes. [16]A population measurement affects the efficiency and performance of GA. GA does poorly for very small size of populations and every average population size influence overall performance of the algorithm. For usual applications, the counseled range is between 10-160 chromosomes. Fitness Evaluate the fitness f(x) of every chromosome x in the population. Create the new population with the aid of repeating following steps till the new population is complete.

Selection: Select two parent chromosomes from a population according to fitness score. Fitness the higher fitness, the bigger chance to be selected.

Crossover: With a crossover probability, move over the parents to structure new offspring (children). If no crossover performed, offspring is the exact copy of parents.

Mutation: With a mutation probability, mutate new offspring at every locus (position in chromosome).
Replace: if the stopping criteria is not satisfied, replace, and use new generated population for next run of the algorithm. If the stopping criteria are satisfied then stop.
Go to step 2 until satisfied solution is coming out.

3. RESULTS AND DISCUSSIONS

In this paper, a well-organized strategy of an obstacle avoidance path planning algorithm for robot hand has been wide-ranging gives rise to a wide-ranging competent manipulator path planning. This chapter presents the individual simulation outcomes for 1, 2, 3 and 4 obstacles avoidance path planning and the performance of the introduced algorithm. Proposed GA path planning is implemented in MATLAB GUI. The case of 3R robot hand arm with initial point (x=0.686m, y=2.268m, θ_g=40º) and final point (x=-2 m, y=-0.5m) is considered while maximum torque for each joint is respectively given as $\tau_{1max} = 50\text{Nm}, \tau_{2max} = 30\text{Nm}$, $\tau_{3max} = 10\text{Nm}$.The parameters of robot hand arm are predefined as $l_1 = 1.2\text{m}$, $l_2 = 0.9\text{m}$, $l_3 = 0.7\text{m}$, $\text{m}_1 = 1.1\text{kg}$, $\text{m}_2 = 0.8\text{kg}$, $\text{m}_3 = 0.6\text{kg}$. The joint velocities and accelerations for the initial and final position are assumed as zeros, and all joints are assumed to rotate 2π freely. The parameters of GA are given as Pc=0.85, Pm=0.07, generation Max=200 and population Size=50. The weight factors are set as w_1=2, w_2=1, w_3=3, w_4=4 and, the 4 obstacles have circular shape with radius 0.35 m and their centers are placed at (-0.4, 1.5), (1.5, 0.7), (1.1, -0.9) and (-1.4, 0.5), respectively. The above settings are entered in MATLAB GUI (table 1 & figure 4 -8).

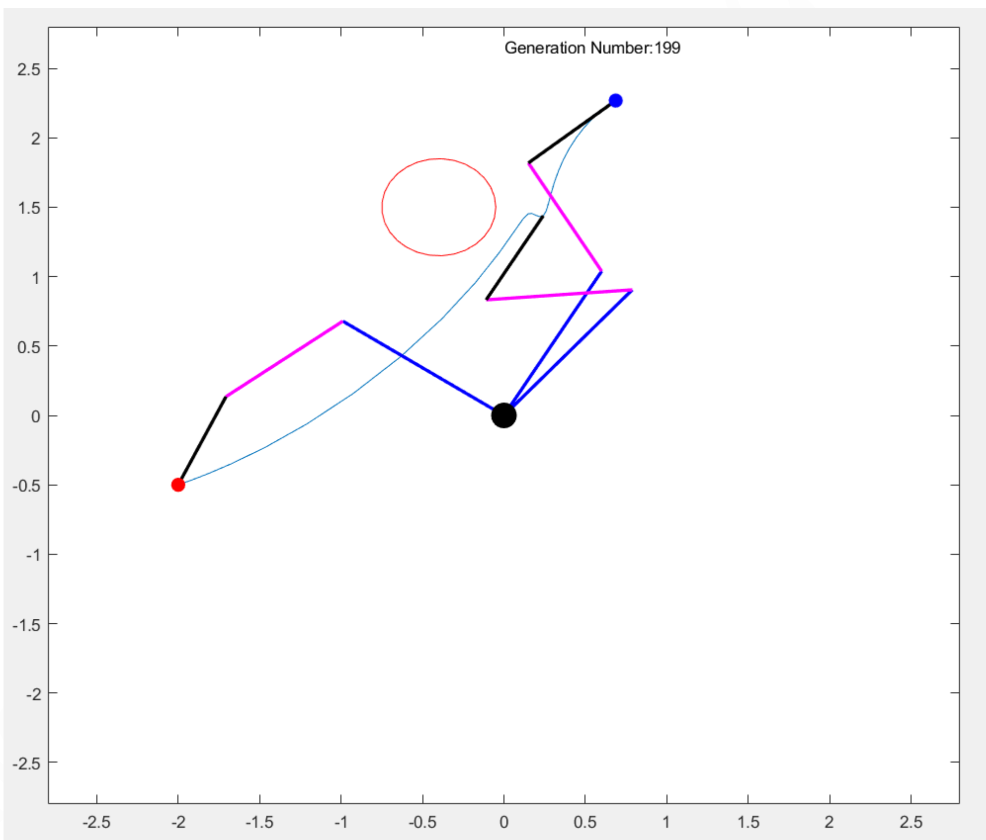


Figure 4 Optimal cartesian path with obstacles

Table 1 Joint angles and torques

Index		Joint 1	Joint 2	Joint 3
Joint torque [Nm][rad]	t=0	11	-2	2
	t=intermediate time	11.5	-3	2.04
	t=final time	-39.3	-2.3	0.3

	maximum absolute angle	11.5	-1.754	2.1
Joint angle [rad]	t=0	1	1	-1.3
	t=intermediate time	1.1	2.5	-2
	t=final time	2.5	1.2	-6.2
	maximum absolute torque	2.5	2.5	-1.3

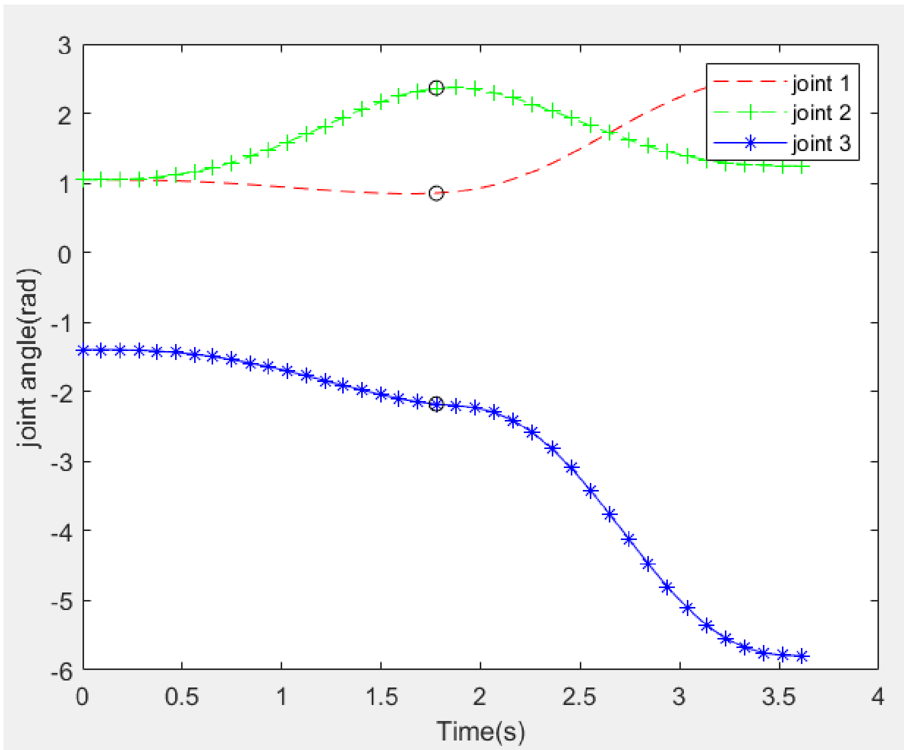


Figure 5 Joint angles versus traveling time

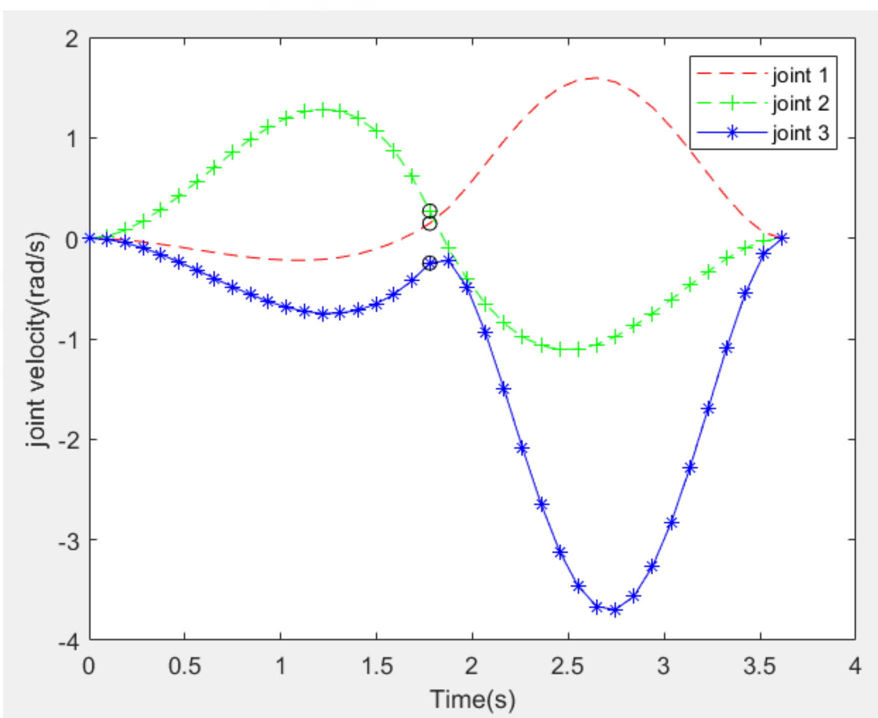


Figure 6 Joint velocities versus traveling time

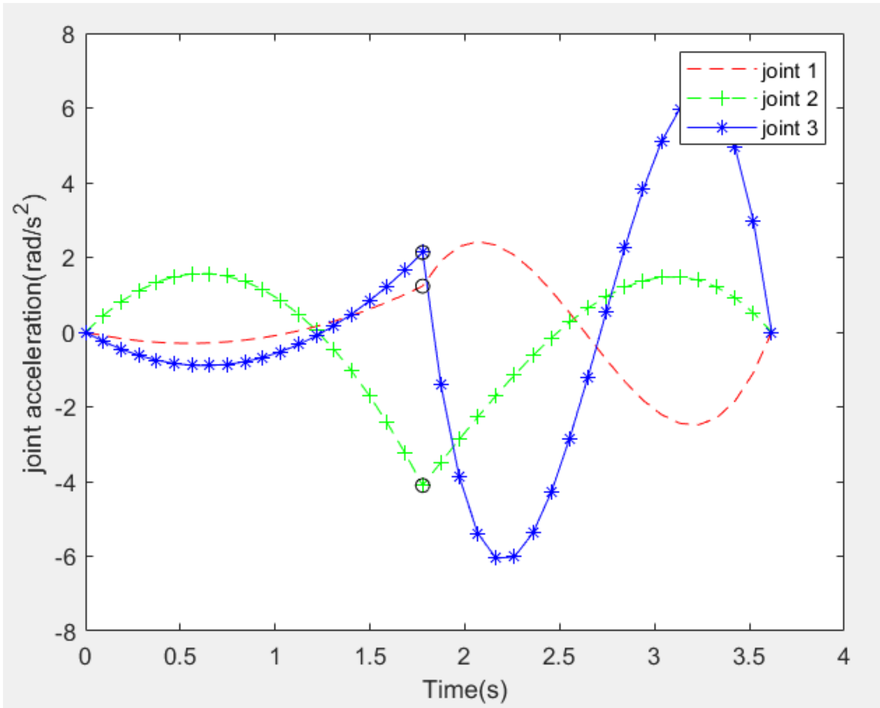


Figure 7 Joint accelerations versus traveling time

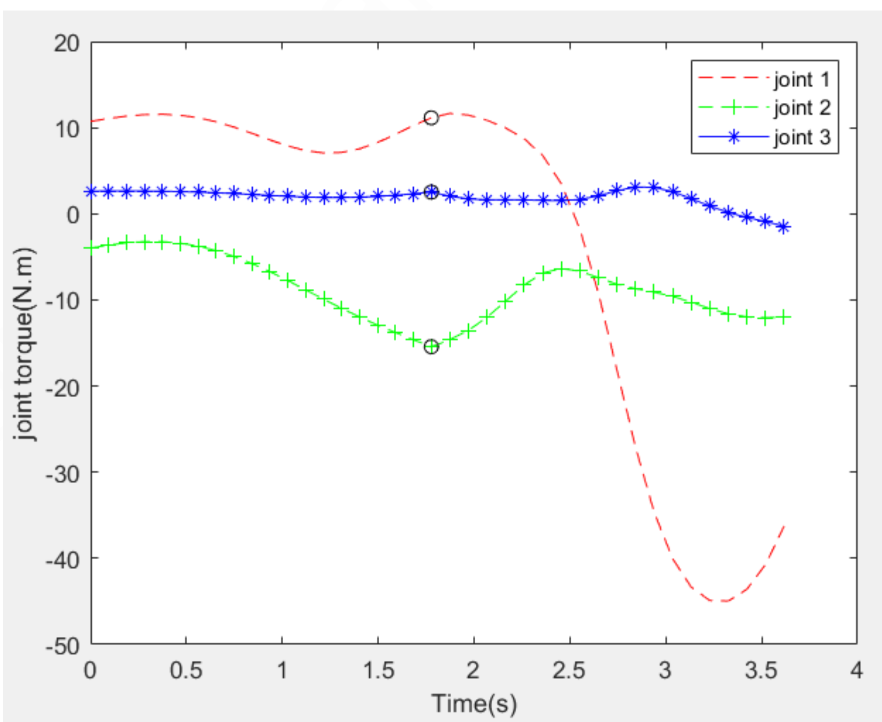


Figure 8 Joint torques versus traveling

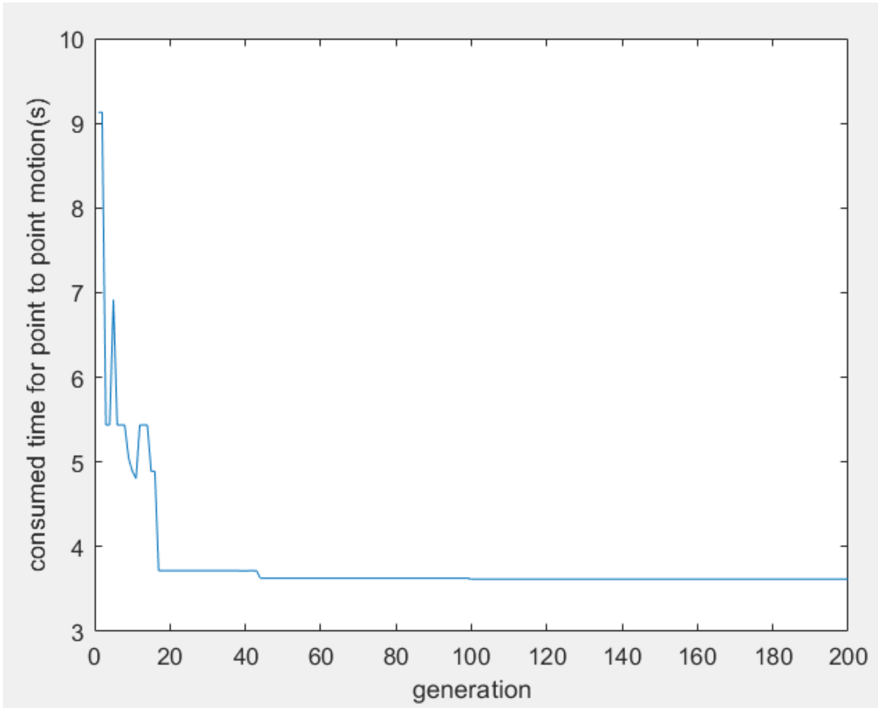


Figure 9 Path traveling time versus generation with one obstacle

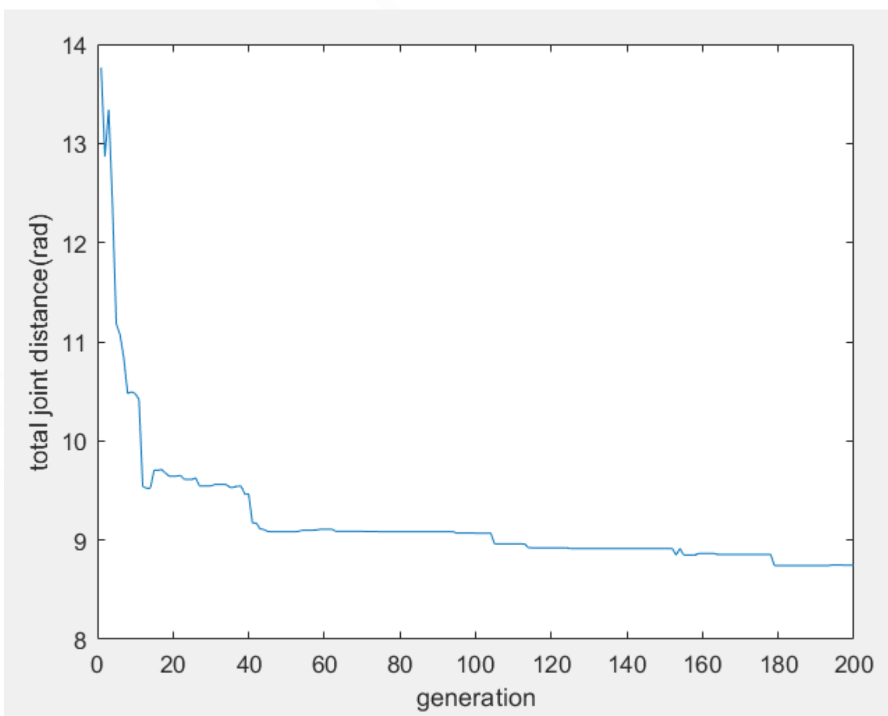
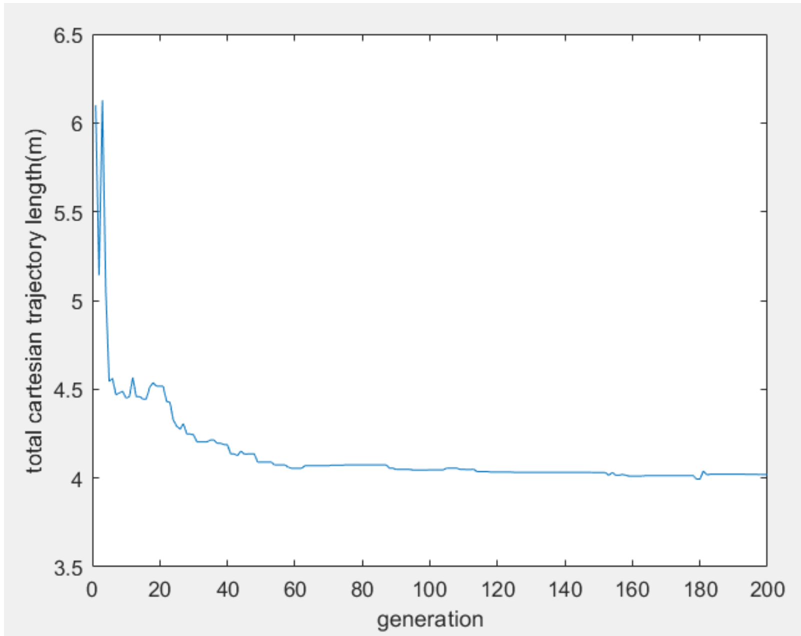


Figure 10 Total joints moving distance versus generation



Figures 11 Total path length versus generation

Figures 9 - 11 are for the convergence of path traveling time, total joints moving distance and path length versus generation max=200. Table 2 shows the optimal path traveling time, total joints moving distance, path length and intermediate time after the optimal generation of GA.

Table 2 Optimal values after final generation

Index	Optimal value
Path traveling time, [s]	3.6
Total joints moving distance, [rad]	8.7
Path length, [m]	4.08
Intermediate time, [s]	1.8

Table 2 shows the least cost path (3.6 m) from start point to goal point while avoiding collisions through one obstacle. The trajectory proposed by GA from the initial configuration to the target configuration found the best path in GA within 4.08 seconds.

Simulation results with two obstacles

This simulation holds two obstacles the aim of the performance is to shows the performance of the proposed obstacle avoidance path planning technique of robot arm. Figures 13 - 20 show the simulation results. The optimized Cartesian path is shown in Figure 12, where the path is not collided with any obstacle. The angle, angular velocity, angular acceleration, and torque for each joint are shown in Figures 14 to 17, where red spots denote the joint angle, angular velocity, angular acceleration, and torque, respectively, at the optimized intermediate time. Especially, the joint torque curves in Figure 17 are not exceeded 50Nm for green curve, 30Nm for blue one and 10Nm for black one, respectively, and these values are set as maximum torque limit for each joint when starting simulation. Table 3 shows the values from Figures 14 to 17.

Table 3 Joint angles and torques

Index		Joint 1	Joint 2	Joint 3
Joint torque [Nm][rad]	t=0	11	-3	0.8
	t=intermediate time	24	-15	-1
	t=final time	-33	-5	-1
	maximum absolute angle	24	-2.5	0.8

Joint angle [rad]	t=0	1.1	1.1	-1.6
	t=intermediate time	0.25	2.2	0.34
	t=final time	2.75	1.75	-1.62
	maximum absolute torque	2.75	2.28	0.38

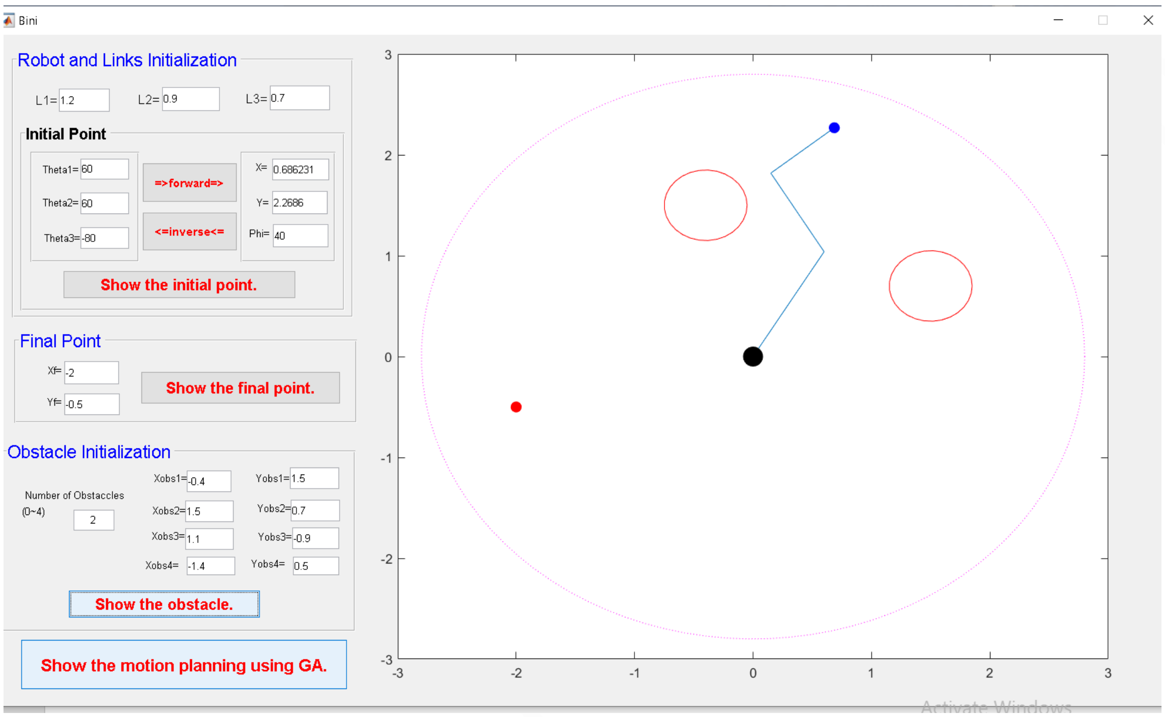


Figure 12 Simulation in matlab

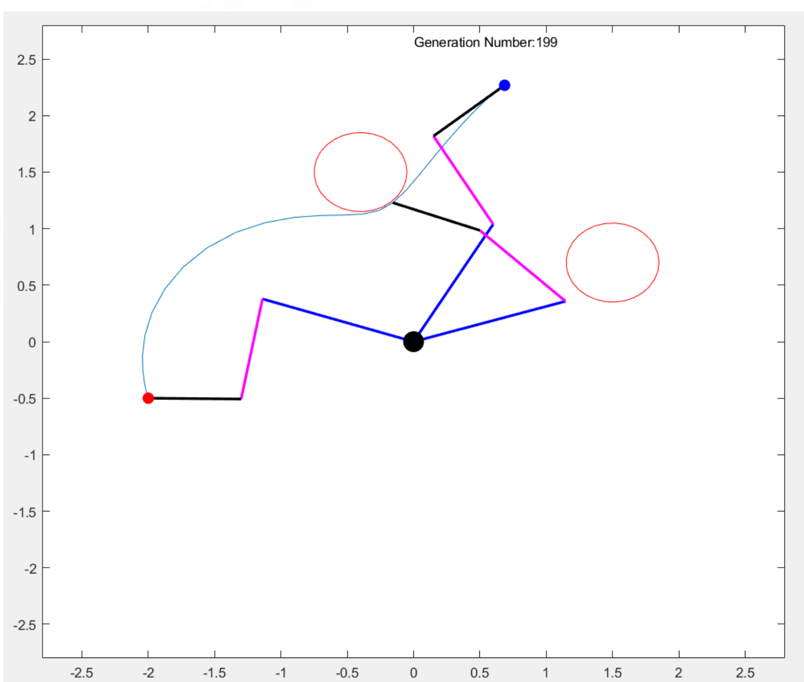


Figure 13 Optimal cartesian path with obstacles

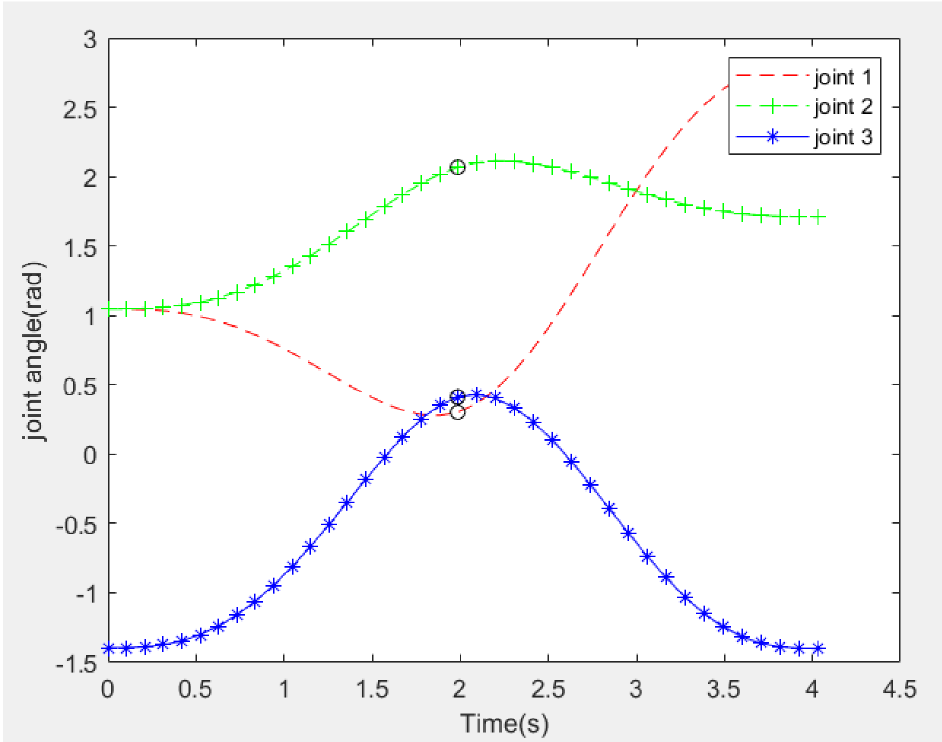


Figure 14 Joint angles versus traveling time

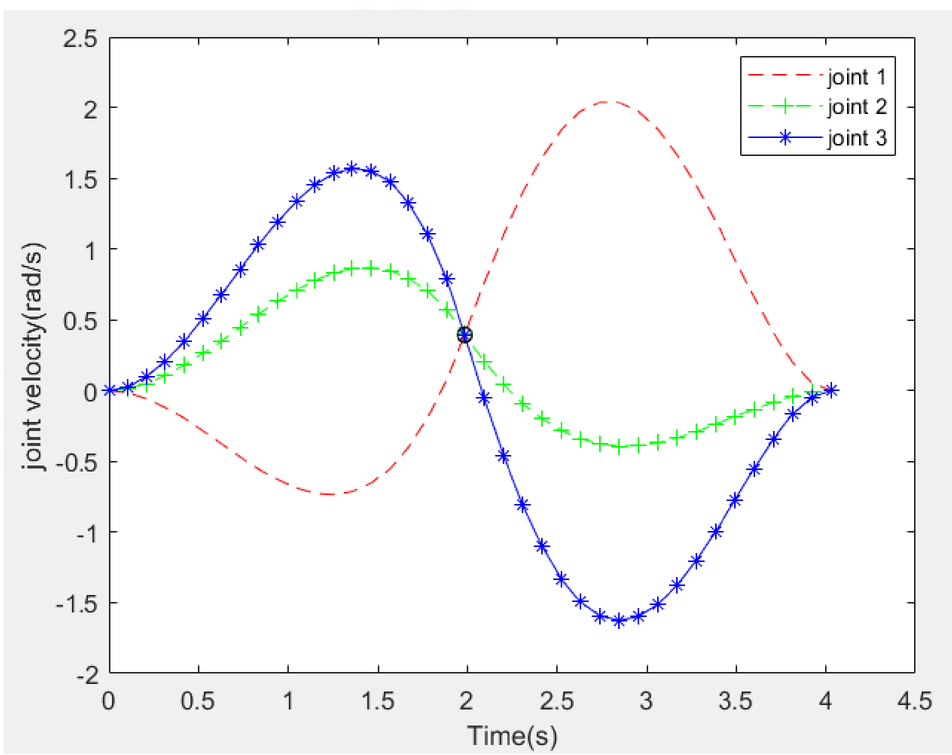


Figure 15 Joint velocities versus traveling time

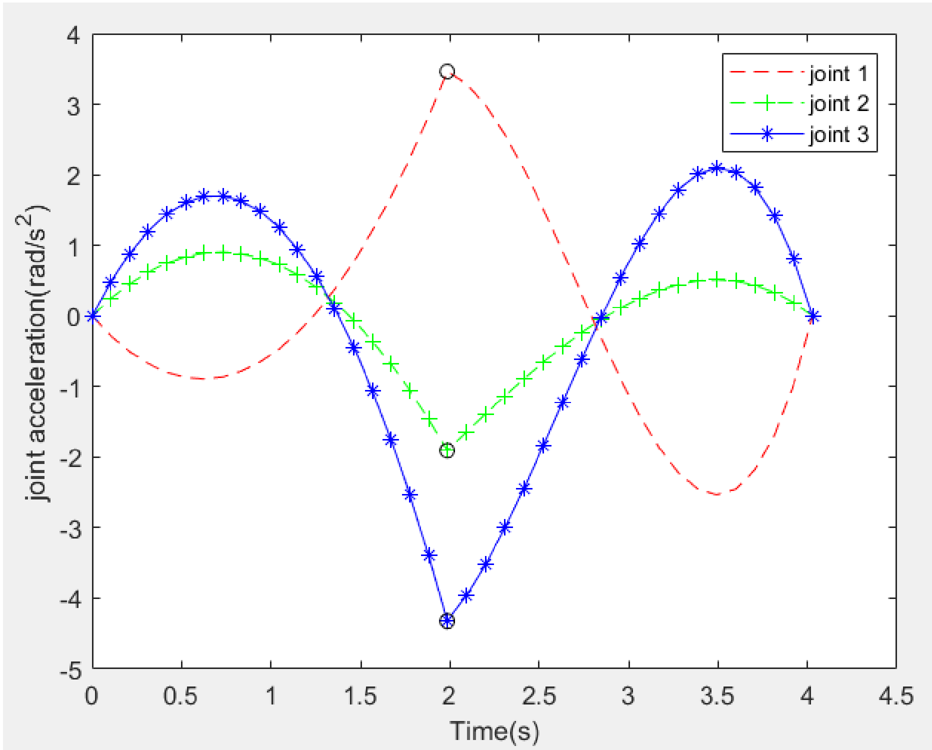


Figure 16 Joint accelerations versus traveling time

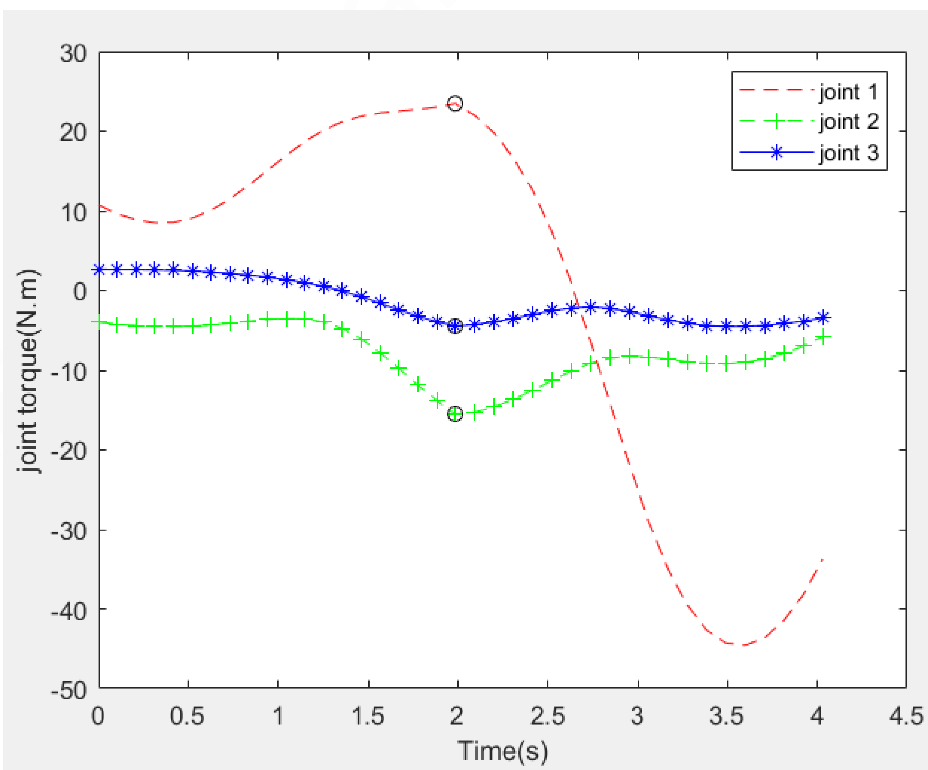


Figure 17 Joint torque versus traveling time

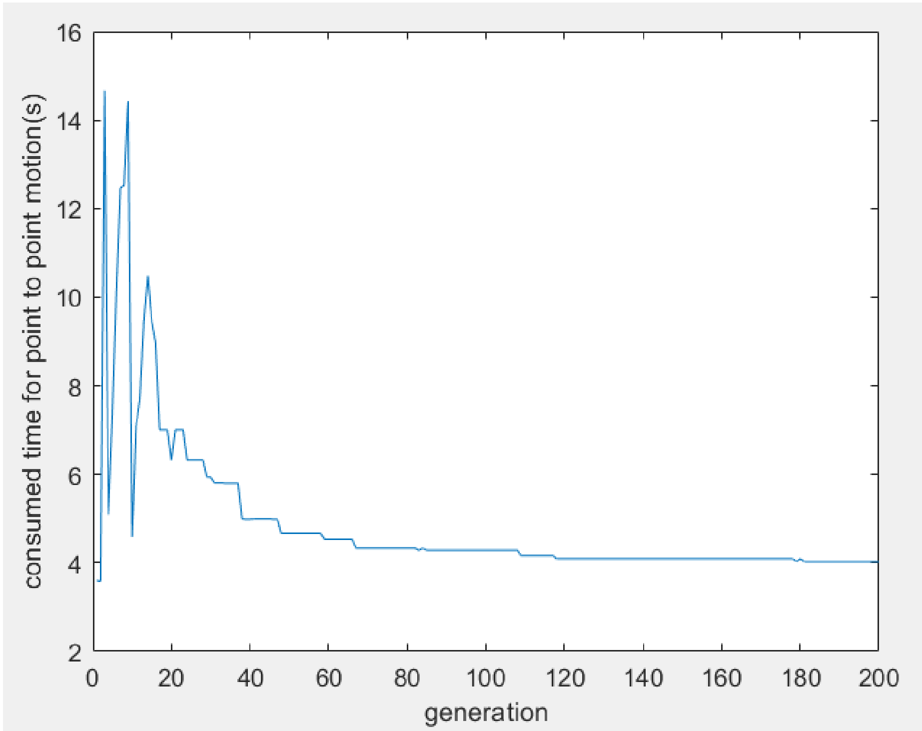


Figure 18 Path traveling time versus generation with two obstacles

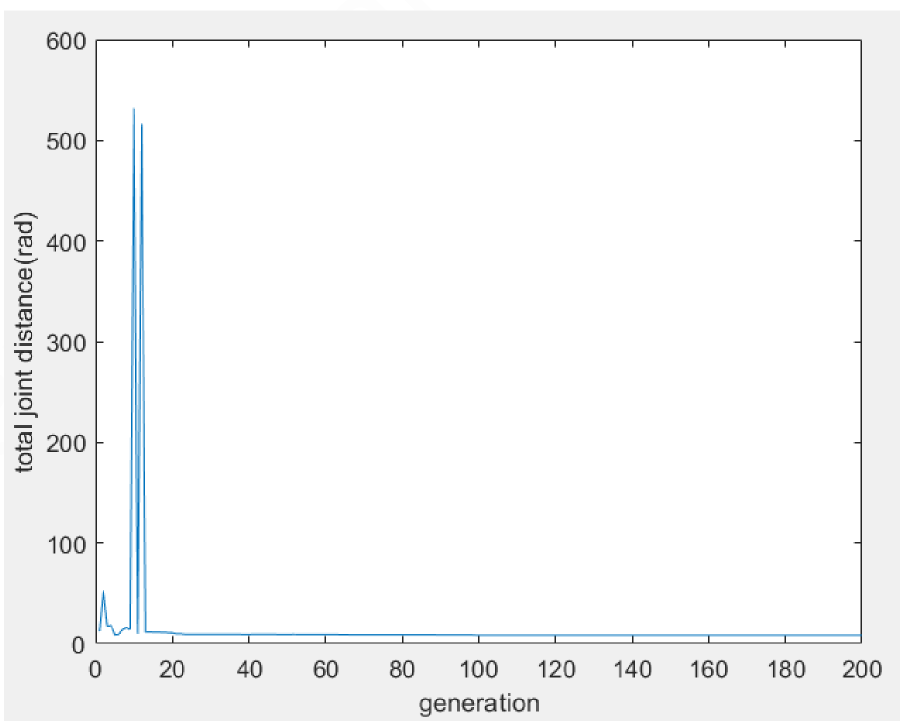


Figure 19 Total joints moving distance versus generation

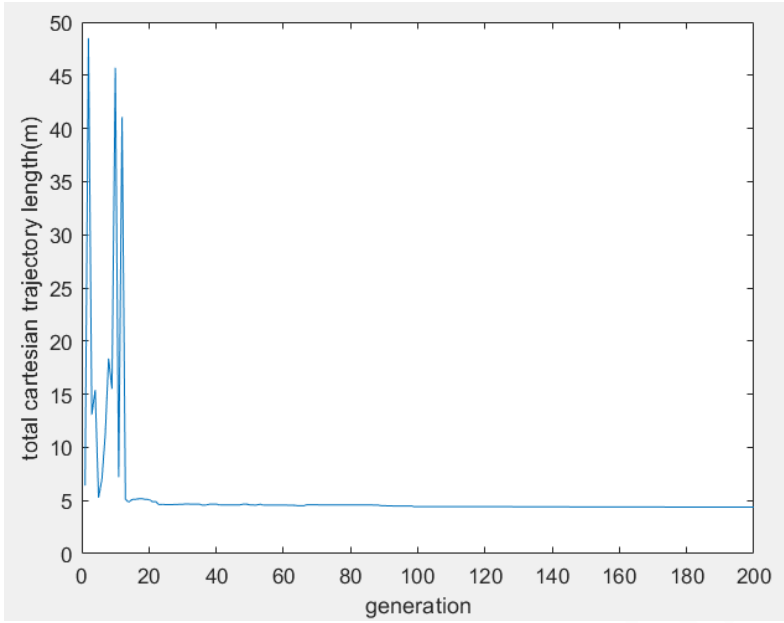


Figure 20 Total path length versus generation

Figures 18 to 20 are for the convergence of path traveling time, total joints moving distance and path length versus generation. The convergence values at generation Max=200 show the optimal one.

4. CONCLUSION

The simulation result leads to the convergence rate and quality of answer in this dissertation, robot hand path planning combined with GA is suggested for less than 200 population size to attain the optimal result. The result is compared to previous work by different researchers it has shown significant improvement and competency concerning time and distance. Forward, inverse kinematics and polynomial path planning strategy of 3R robot arm were studied for redundancy of final configuration and GA approach. The chromosome has been designed and GA operators were applied according to the case of 3R links. The fitness function of GA was designed from the needs that have to find the shortest Cartesian trajectory length, traveling time and joints moving distance while avoiding obstacles and not exceeding maximum torque. The proposed method has been implemented in Matlab software. The simulation results show the validity of this method and satisfy the objectives and constraints.

Despite the significant improvements made, there is more room for improvement. For example, If the GA falls in the local optimal point it can't go further to search for global optimal which is clearly required by way of us. This property of falling within the local best space is acknowledged as convergence Currently, I am working with my supervisor the third project on a combination of the Bayesian network with GA to improve the efficiently in the trajectory of the robot arm to reach the optimal value.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

REFERENCE

1. Zahra, Kiran, Frank O. Ostermann, and Ross S. Purves. "Geographic variability of Twitter usage characteristics during disaster events." Geo-spatial information science 20.3 (2017): 231-240.
2. Hassani, Aydin, et al. "Preparation of magnetite nanoparticles by high-energy planetary ball mill and its application for ciprofloxacin degradation through heterogeneous Fenton process." Journal of environmental management 211 (2018): 53-62.
3. Kumar, Santosh, et al. "Catecholamine-induced myocardial fibrosis and oxidative stress is attenuated by Terminalia arjuna (Roxb.)." Journal of Pharmacy and Pharmacology 61.11 (2009): 1529-1536.
4. Bejan, Adrian, George Tsatsaronis, and Michael Moran. Thermal design and optimization. John Wiley & Sons, 1995.

5. Zhu, D., et al. "Transfection of C6 glioma cells with connexin 43 cDNA: analysis of expression, intercellular coupling, and cell proliferation." Proceedings of the National Academy of Sciences 88.5 (1991): 1883-1887.
6. Atyabi, Adham, and SamiaNefti-Meziani. "Applications of computational intelligence to robotics and autonomous systems." Handbook on Computational Intelligence: Volume 2: Evolutionary Computation, Hybrid Systems, and Applications. 2016. 821-863.
7. Lingelbach, Frank. "Path planning using probabilistic cell decomposition."IEEE International Conference on Robotics and Automation, 2004.Proceedings.ICRA'04. 2004. Vol. 1. IEEE, 2004.
8. Rosell, Jan, and Pedro Iniguez. "Path planning using harmonic functions and probabilistic cell decomposition."Proceedings of the 2005 IEEE international conference on robotics and automation. IEEE, 2005.
9. Balzarini, Rhonda N., et al. "Does exposure to erotica reduce attraction and love for romantic partners in men? Independent replications of Kenrick, Gutierres, and Goldberg (1989) study 2." Journal of Experimental Social Psychology 70 (2017): 191-197.
10. Steele, A. D., et al. "Anticipating rotavirus vaccines: epidemiology and surveillance of rotavirus in South Africa." Vaccine 21.5-6 (2003): 354-360.
11. Kozek, Tibor, TamásRoska, and Leon O. Chua. "Genetic algorithm for CNN template learning." IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications 40.6 (1993): 392-402.
12. Khanbary, Lutfi Mohammed Omer, and Deo Prakash Vidyarthi. "A GA-based effective fault-tolerant model for channel allocation in mobile computing." IEEE Transactions on Vehicular Technology 57.3 (2008): 1823-1833.
13. Younes, Mimoun, Fouad Khodja, and Riad Lakhdar Kherfane. "Multi-objective economic emission dispatch solution using hybrid FFA (firefly algorithm) and considering wind power penetration." Energy 67 (2014): 595-606.
14. Eiben, Ágoston E., Robert Hinterding, and Zbigniew Michalewicz. "Parameter control in evolutionary algorithms." IEEE Transactions on evolutionary computation 3.2 (1999): 124-141.
15. The initial population of chromosomes can be generated at random in the predefined interval of each chromosome, and the search is then carried out among this population
16. Petrovski, Andrei, Alexander Brownlee, and John McCall. "Statistical optimisation and tuning of GA factors."2005 IEEE Congress on Evolutionary Computation. Vol. 1. IEEE, 2005.